



LOOPING STATEMENTS

Dhruvin Dholiya



1. Write C program to print 1 to 10 number.

❖ Code:

```
#include<stdio.h>

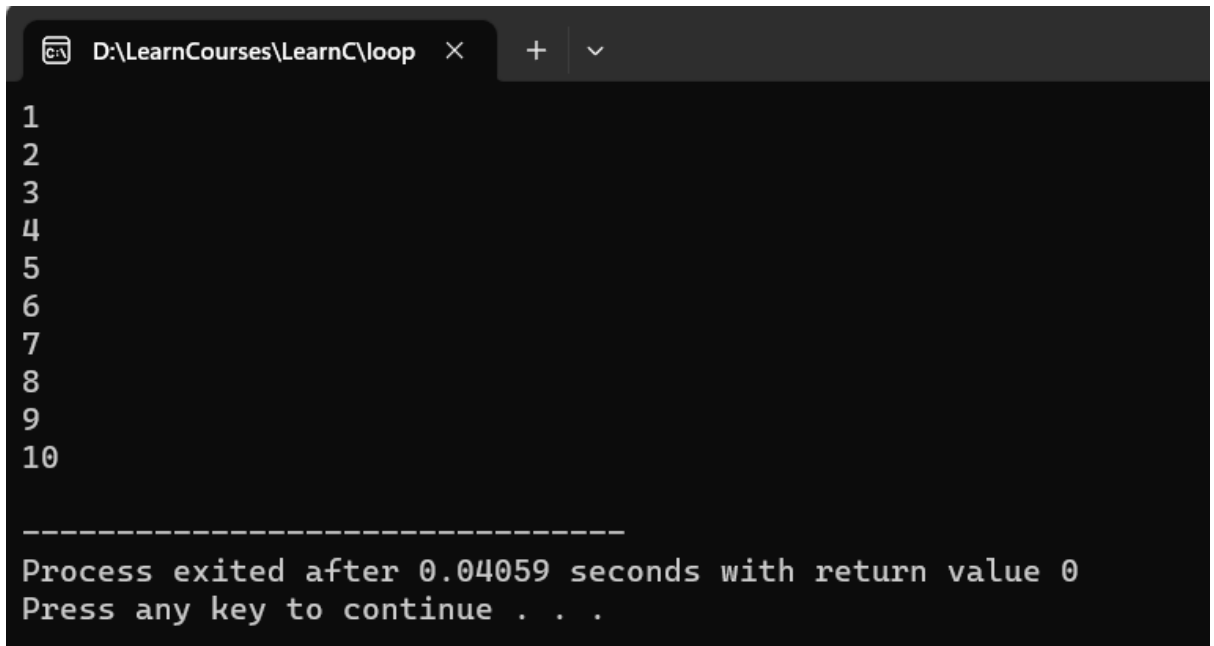
int main() {

    int i;

    for (i=1; i<= 10; i++) {
        printf("%d\n", i);
    }

    return 0;
}
```

❖ Output:

A screenshot of a Windows command prompt window. The title bar shows the file path 'D:\LearnCourses\LearnC\loop' and standard window controls. The command prompt displays the output of a C program: the numbers 1 through 10, each on a new line. Below the numbers, a separator line is followed by the text 'Process exited after 0.04059 seconds with return value 0' and 'Press any key to continue . . .'.

```
D:\LearnCourses\LearnC\loop >
1
2
3
4
5
6
7
8
9
10

-----
Process exited after 0.04059 seconds with return value 0
Press any key to continue . . .
```

2. Write C program to print 40 to 31 number.

❖ Code:

```
#include<stdio.h>

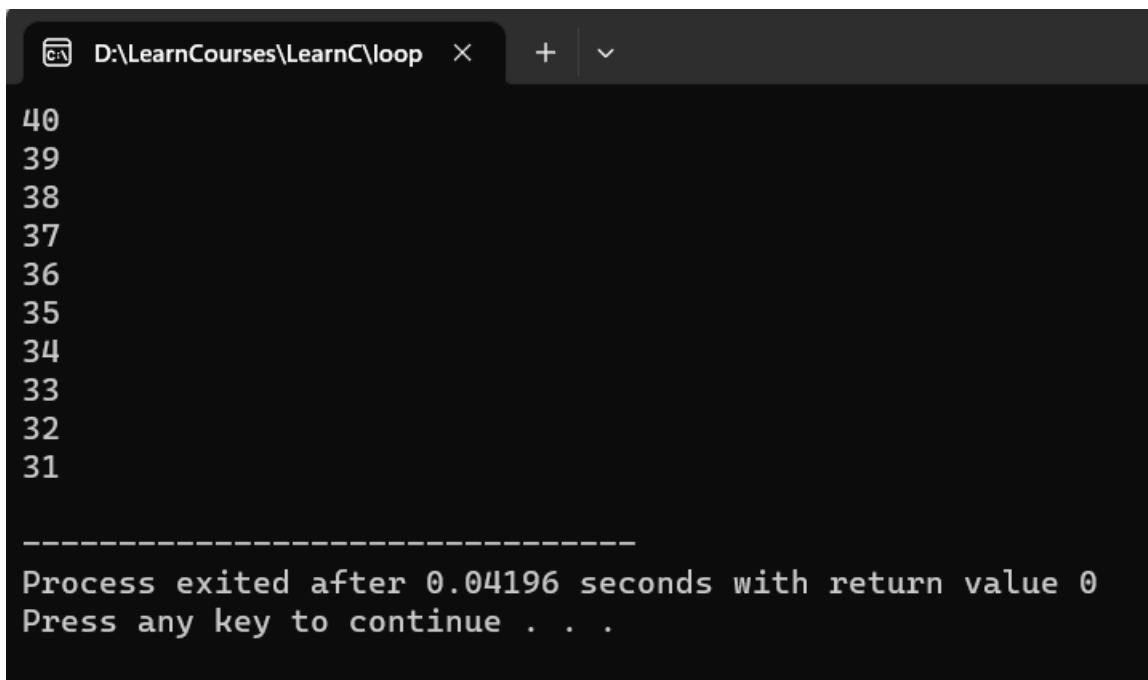
int main() {

    int i;

    for (i=40; i>= 31; i--) {
        printf("%d\n", i);
    }

    return 0;
}
```

❖ Output:

A screenshot of a terminal window with a dark background. The window title bar shows the file path 'D:\LearnCourses\LearnC\loop' and standard window controls. The output of the program is displayed in white text, showing a list of numbers from 40 down to 31, each on a new line. Below the numbers, a separator line of dashes is followed by the message 'Process exited after 0.04196 seconds with return value 0' and 'Press any key to continue . . .'.

```
D:\LearnCourses\LearnC\loop  X  +  v

40
39
38
37
36
35
34
33
32
31

-----
Process exited after 0.04196 seconds with return value 0
Press any key to continue . . .
```

3. Write C program to print odd number from 80 to 100.

❖ Code:

```
#include<stdio.h>

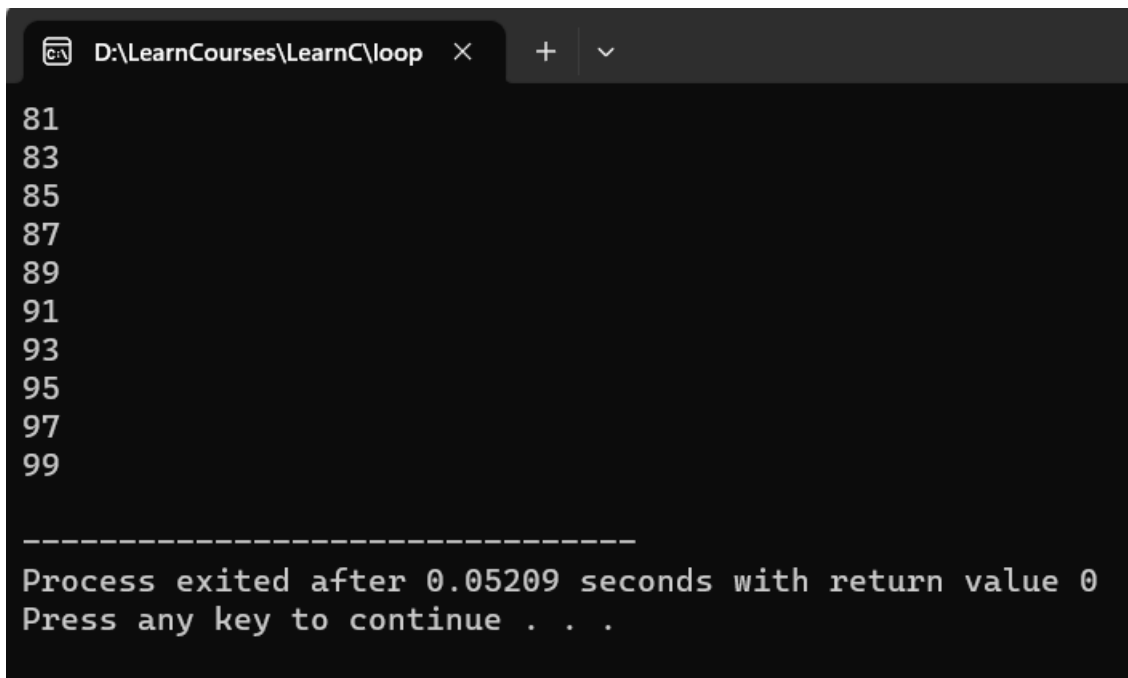
int main() {

    int i;

    for (i=80; i<= 100; i++) {
        if (i % 2 == 1) {
            printf("%d\n", i);
        }
    }

    return 0;
}
```

❖ Output:

A screenshot of a terminal window with a dark background. The window title bar shows a file icon, the path 'D:\LearnCourses\LearnC\loop', and standard window controls (close, maximize, minimize). The terminal displays the output of the C program: odd numbers from 81 to 99, each on a new line. Below the numbers, there is a dashed line followed by the text 'Process exited after 0.05209 seconds with return value 0' and 'Press any key to continue . . .'.

```
D:\LearnCourses\LearnC\loop  ×  +  ▾

81
83
85
87
89
91
93
95
97
99

-----
Process exited after 0.05209 seconds with return value 0
Press any key to continue . . .
```

4. Write C program to print number that are divisible by 5 from 25 to 50 number.

❖ Code:

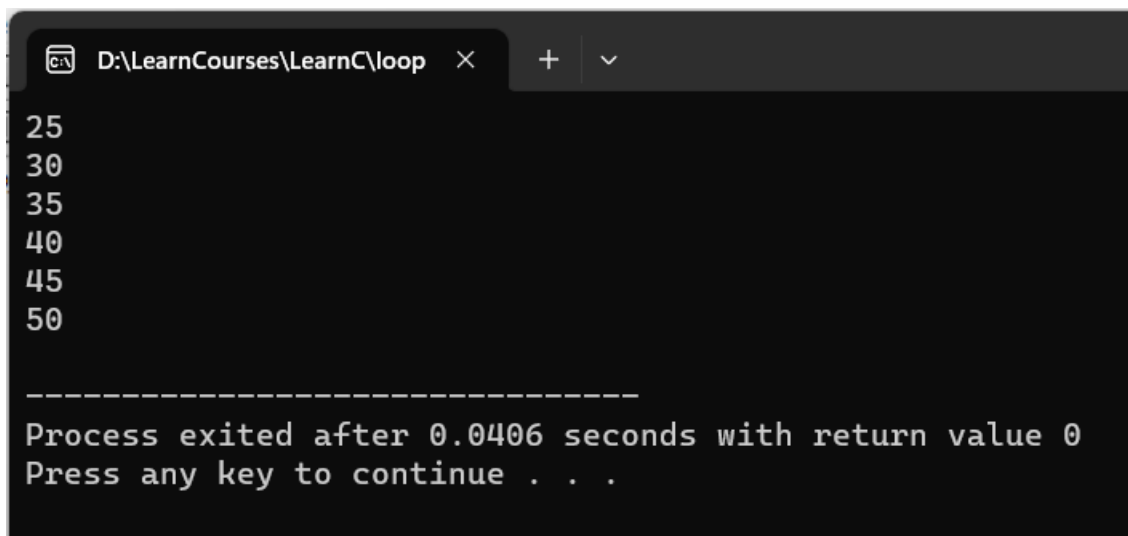
```
#include<stdio.h>
int main() {

    int i;

    for (i=25; i<= 50; i+=5) {
        printf("%d\n", i);
    }

    return 0;
}
```

❖ Output:



The screenshot shows a Windows command prompt window with the title bar "D:\LearnCourses\LearnC\loop". The output of the program is displayed as follows:

```
25
30
35
40
45
50

-----
Process exited after 0.0406 seconds with return value 0
Press any key to continue . . .
```

5. Write C program to print A to Z character.

❖ Code:

```
#include<stdio.h>

int main() {

    char i;

    for (i='A'; i<='Z'; i++) {
        printf("%c\n", i);
    }

    return 0;
}
```

❖ Output:



```
D:\LearnCourses\LearnC\loop X + v
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

-----
Process exited after 0.04169 seconds with return value 0
Press any key to continue . . . |
```

6. Write C program to print multiplication table of any number.

❖ Code:

```
#include<stdio.h>

int main() {

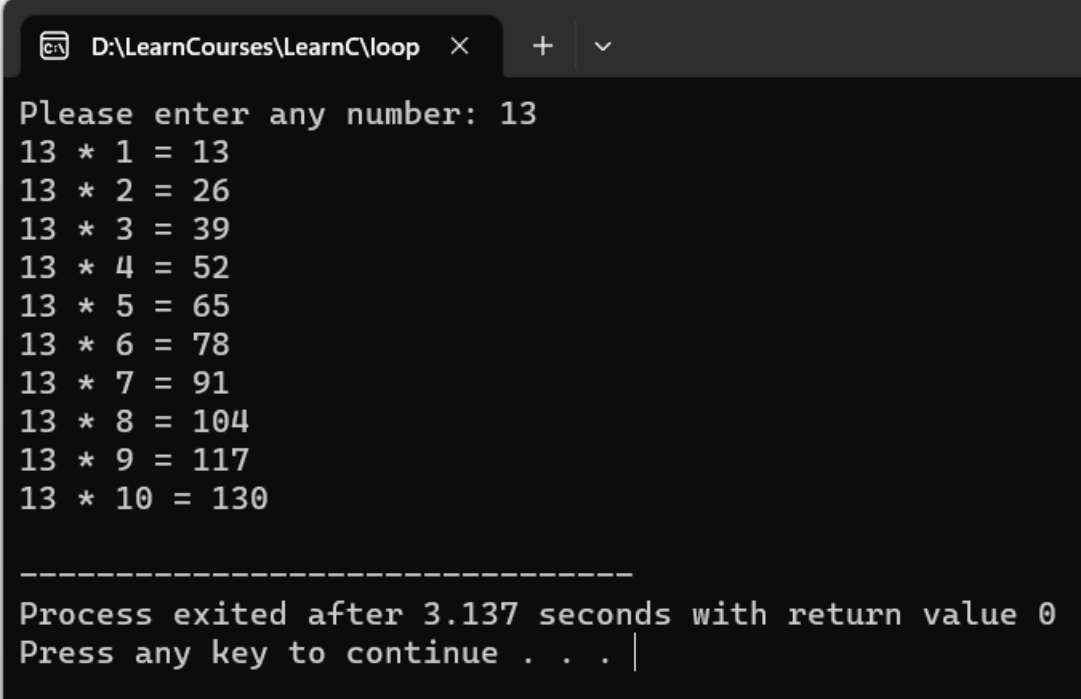
    int i, num;

    printf("Please enter any number: ");
    scanf("%d", &num);

    for (i=1; i<=10; i++) {
        printf("%d * %d = %d\n", num, i, num*i);
    }

    return 0;
}
```

❖ Output:

A screenshot of a Windows command prompt window. The title bar shows the file path 'D:\LearnCourses\LearnC\loop' and standard window controls. The prompt displays the output of the C program: it first asks 'Please enter any number:' and receives '13'. It then prints a multiplication table for 13, with rows from '13 * 1 = 13' to '13 * 10 = 130'. Below the table, a separator line is followed by the message 'Process exited after 3.137 seconds with return value 0' and 'Press any key to continue . . . |'.

```
D:\LearnCourses\LearnC\loop  ×  +  ∨

Please enter any number: 13
13 * 1 = 13
13 * 2 = 26
13 * 3 = 39
13 * 4 = 52
13 * 5 = 65
13 * 6 = 78
13 * 7 = 91
13 * 8 = 104
13 * 9 = 117
13 * 10 = 130

-----
Process exited after 3.137 seconds with return value 0
Press any key to continue . . . |
```


7. Write C program to count number of boys whose weight is less than 50kg and height is greater than 170cm.

❖ **Code:**

```
#include<stdio.h>
int main() {

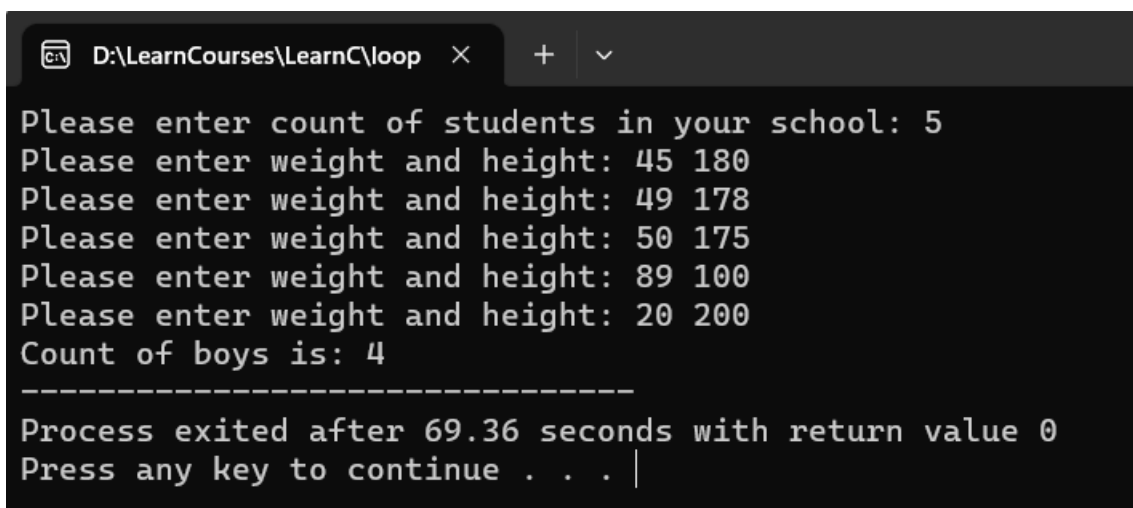
    int i, count=0, students;
    float height, weight;

    printf("Please enter count of students in your school: ");
    scanf("%d", &students);

    for (i=1; i<=students; i++) {
        printf("Please enter weight and height: ");
        scanf("%f %f", &weight, &height);
        if (weight <= 50 && height >= 170) {
            count++;
        }
    }

    printf("Count of boys is: %d", count);
    return 0;
}
```

❖ **Output:**

A screenshot of a Windows command prompt window titled "D:\LearnCourses\LearnC\loop". The window shows the execution of a C program. The user enters '5' for the number of students. Then, five pairs of weight and height values are entered: (45, 180), (49, 178), (50, 175), (89, 100), and (20, 200). The program outputs "Count of boys is: 4". Below this, it shows "Process exited after 69.36 seconds with return value 0" and "Press any key to continue . . . |".

```
D:\LearnCourses\LearnC\loop >
Please enter count of students in your school: 5
Please enter weight and height: 45 180
Please enter weight and height: 49 178
Please enter weight and height: 50 175
Please enter weight and height: 89 100
Please enter weight and height: 20 200
Count of boys is: 4
-----
Process exited after 69.36 seconds with return value 0
Press any key to continue . . . |
```

8. Write C program to find ratio of (a-b) and (c-d) of any three number. If c and d is equal than not to find ratio.

❖ Code:

❖ Output:

9. Write C program to calculate factorial of a number.

❖ Code:

```
#include<stdio.h>
int main() {

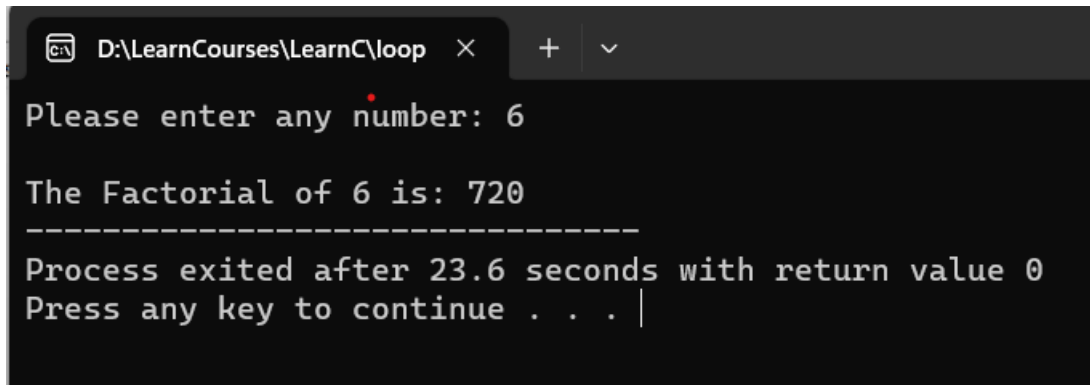
    int i, num, factorial;

    printf("Please enter any number: ");
    scanf("%d", &num);

    for (i=1; i<=num; i++) {
        factorial = factorial*i;
    }
    printf("\nThe Factorial of %d is: %d", num, factorial);

    return 0;
}
```

❖ Output:

A screenshot of a Windows command prompt window. The title bar shows the file path 'D:\LearnCourses\LearnC\loop'. The prompt displays the text 'Please enter any number: 6' followed by a red cursor. Below that, it shows 'The Factorial of 6 is: 720'. A dashed line separates this from the final output: 'Process exited after 23.6 seconds with return value 0' and 'Press any key to continue . . . |' with a vertical cursor.

```
D:\LearnCourses\LearnC\loop >
Please enter any number: 6
The Factorial of 6 is: 720
-----
Process exited after 23.6 seconds with return value 0
Press any key to continue . . . |
```

10. Write C program to give sum of n numbers.

❖ Code:

```
#include<stdio.h>
int main() {

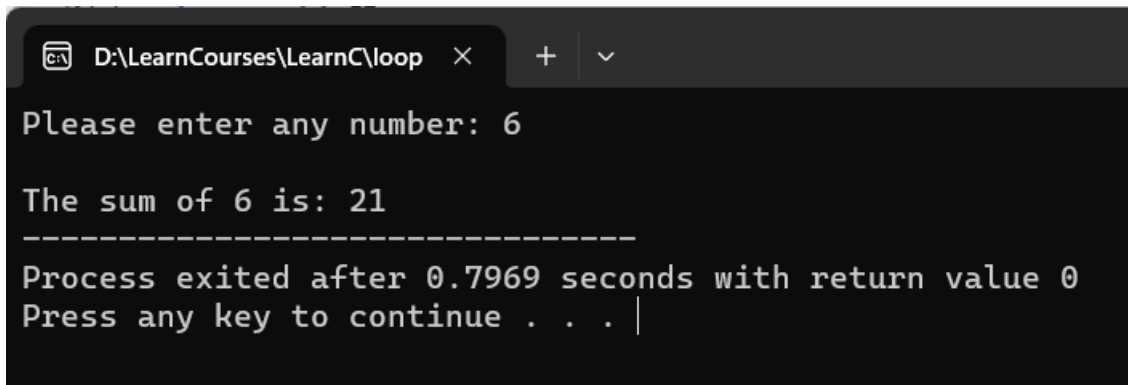
    int i, num, sum=0;

    printf("Please enter any number: ");
    scanf("%d", &num);

    for (i=1; i<=num; i++) {
        sum = sum + i;
    }
    printf("\nThe sum of %d is: %d", num, sum);

    return 0;
}
```

❖ Output:

A screenshot of a terminal window with a dark background. The window title bar shows a file icon, the path 'D:\LearnCourses\LearnC\loop', and standard window controls (close, maximize, minimize). The terminal output shows the program's execution: it prompts 'Please enter any number: 6', then displays 'The sum of 6 is: 21'. Below this, a dashed line separates the output from a status message: 'Process exited after 0.7969 seconds with return value 0'. The prompt 'Press any key to continue . . . |' is visible at the bottom.

```
D:\LearnCourses\LearnC\loop × + ▾
Please enter any number: 6
The sum of 6 is: 21
-----
Process exited after 0.7969 seconds with return value 0
Press any key to continue . . . |
```

11. Write C program to give average of n numbers.

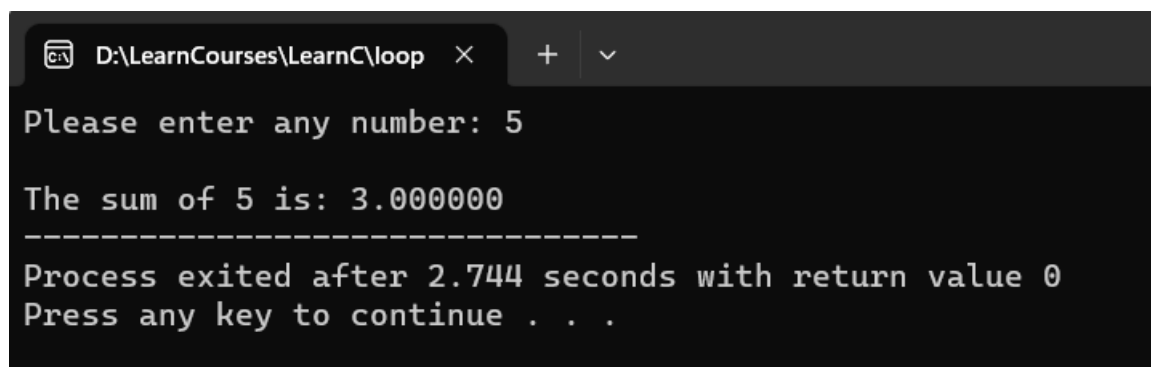
❖ Code:

```
#include<stdio.h>

int main() {
    int i, num;
    float avg;
    printf("Please enter any number: ");
    scanf("%d", &num);
    for (i=1; i<=num; i++) {
        avg = avg + i;
    }
    avg = avg / num;
    printf("\nThe sum of %d is: %f", num, avg);

    return 0;
}
```

❖ Output:

A screenshot of a terminal window with a dark background. The window title bar shows a file icon, the path 'D:\LearnCourses\LearnC\loop', and standard window controls. The terminal output shows the program's execution: it prompts 'Please enter any number: 5', calculates the average, and outputs 'The sum of 5 is: 3.000000'. Below this, a separator line is followed by the message 'Process exited after 2.744 seconds with return value 0' and 'Press any key to continue . . .'.

```
D:\LearnCourses\LearnC\loop × + ▾
Please enter any number: 5

The sum of 5 is: 3.000000
-----
Process exited after 2.744 seconds with return value 0
Press any key to continue . . .
```